SPRY SOFTWARE TRIP USER MANUAL

TABLE OF CONTENTS

Installation and Updating 2

Installation from an Eclipse Update Site 2

Installation from a JAR file 2

Updating Trip after installation 3

Using Trip 4

Trip Rationale 4

Create a New Trip Diagram 4

Arranging and Viewing the Diagram 4

Undo and Redo 5

Add a New Class or Interface 5

Add a New Method 5

Add a New Field 6

Modification of Types, Methods and Fields 6

Delete Elements 6

Create a New Relationship 7

Relationship Routing 7

Moving Relationships 7

Filter Visible Relationships 8

Committing Changes 8

Real-Time Mode 8

Saving the Diagram 8

Diagram Context 8

Jump to Code 9

The Source Code 10

Importing the Source Code into Eclipse 10

Building the Source Code in Eclipse 11

Running Trip Unit Tests 11

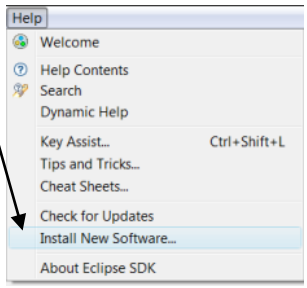
Known Issues 14

Installation and Updating

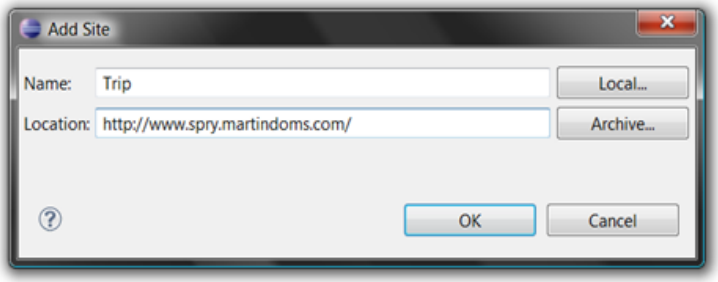
This section will guide you through installation of the Trip plug-in.

Installation from an Eclipse Update Site

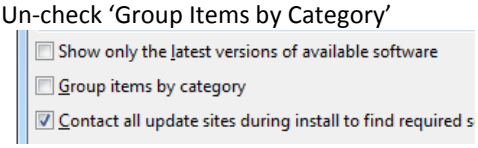
1. Select **Help>Install New Software** to bring up the install dialog



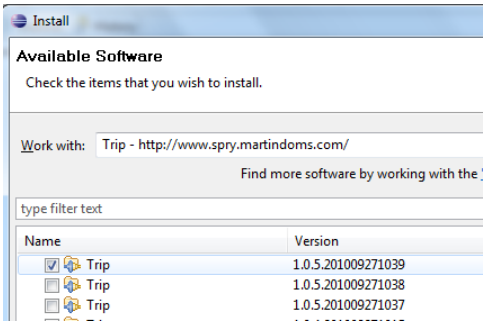
2. Click **Add** and fill in the Trip update site details as shown in the diagram

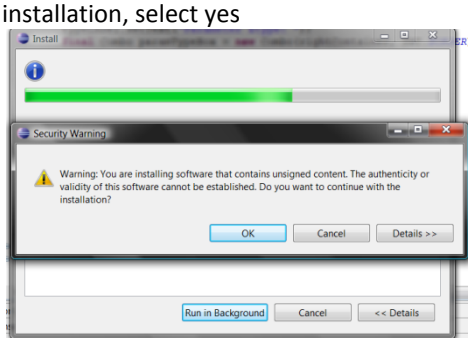


3. Un-check 'Group Items by Category'

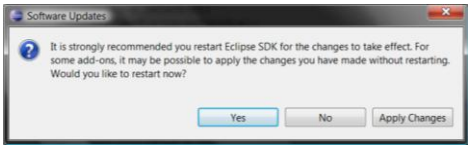

Show only the latest versions of available software
Group items by category
Contact all update sites during install to find required s

4. Select the latest version of the Trip plug-in and click **Next**
5. Click **Finish** on the dialog that appears
6. You may be prompted to accept the terms of the license agreement. Click **Finish**
7. If you are prompted to confirm the unsigned plug-in installation, select yes




Security Warning
Warning: You are installing software that contains unsigned content. The authenticity or validity of this software cannot be established. Do you want to continue with the installation?
OK
Cancel
Details >>
Run in Background
Cancel
<< Details

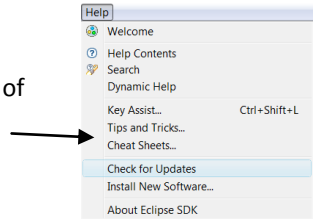
8. Installation time varies depending on how many other required plug-ins need to be downloaded
9. Restart Eclipse once the plug-in installs
10. Trip is now ready to be used


Software Updates
It is strongly recommended you restart Eclipse SDK for the changes to take effect. For some add-ons, it may be possible to apply the changes you have made without restarting. Would you like to restart now?
Yes
No
Apply Changes

Installation from a JAR file

1. Download the Trip.jar file. This is located on the Final Release page of the wiki (http://redmine.cosc.canterbury.ac.nz:18080/redmine/wiki/2010-c325g3/Final_Release)
2. Move this into your Eclipse plug-ins folder
3. Restart Eclipse if it is running
4. Trip is now ready to use

UPDATING TRIP AFTER INSTALLATION



1. Select **Help>Check** for Updates
2. Eclipse will contact the Trip update site and will return the latest version of the plug-in
3. Select Trip from the list and click **Next**
4. You may be prompted to confirm the plug-in installation
5. Once the plug-in installs, Eclipse may prompt you to restart
6. Trip is now updated and ready to be used

USING TRIP

TRIP RATIONALE

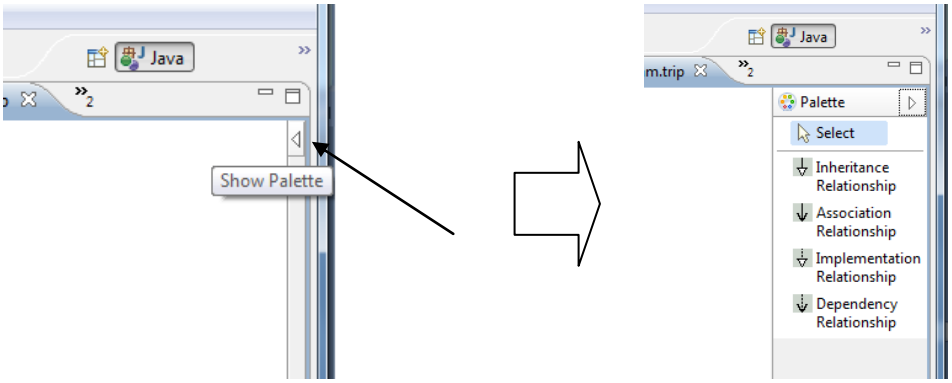
Trip operates using a transactional model therefore changes made to a Trip diagram do not immediately appear in code. Changes are coloured in such a way that they can be easily seen on the diagram and are written to code by way of the Commit Changes button.

CREATE A NEW TRIP DIAGRAM

- Open your Java project or create a new project
- Click File > New > Other... (Ctrl + N)
- Under Trip Diagram Tools, select Class Diagram, click Next
- In Container, enter the project name (this is case sensitive)
- You can enter a suitable name for the diagram in File Name. Include the .trip extension
- If you would like all existing classes in the project to be imported into the diagram, select 'Full Project'. Otherwise select 'Empty Diagram'
- Click Finish
- The diagram will open automatically

ARRANGING AND VIEWING THE DIAGRAM

- To begin, expand the palette. You can do so by clicking this arrowhead



- Use the Select tool from the palette (see below) to select and move diagram elements
- Single elements may be selected and moved individually by clicking and holding the left mouse button anywhere inside them
- Elements can be dragged off the diagram and the canvas grows to accommodate
- Pause holding an element right at the edge of the visible diagram and the diagram will pan away revealing clear space. Drop it and then drag it into the precise spot
- Multiple elements may be selected by:
    - Holding the 'ctrl' key and clicking on each or
    - Dragging a selection box over the desired items
- They can then be dragged around as a group
- Hold ctrl and spin the mouse wheel to zoom in and out
- The palette can be moved to the left or right of the diagram by left clicking and dragging it by the "Palette" text or by right clicking on Palette and selecting Dock on > Left or Dock on > Right
- The palette can be resized by using the handle that appears when you hover the mouse in-between the scroll bar and Palette or by right clicking on the Palette text and selecting Resize

- You can right click anywhere else in the palette to change Layout, Icons and other settings such as fonts

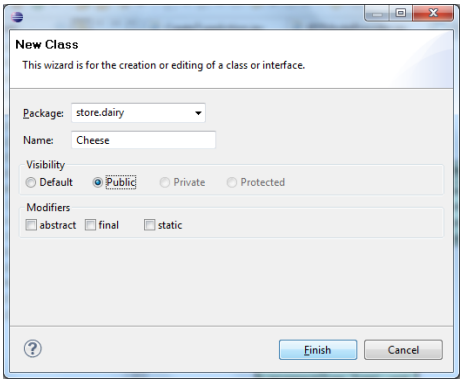
Undo and Redo

The Spry Software team is committed to making Trip a safe environment to work in. This explains the transactional model and undo and redo that goes beyond the 'Commit Changes' button. If, for example, a Class is marked for deletion and the change is then committed you can revert to the earlier state by:

- Click the 'Undo' button (ctrl+z) and the file is reinstated
- Click it again and the class is unmarked
- Click the 'Redo' button (ctrl+y) and the file is marked again
- Click it again and the file is deleted

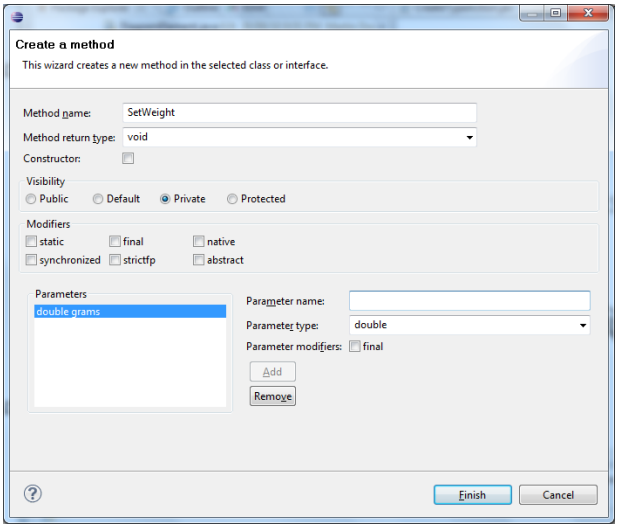
Add a New Class or Interface

- Right click on the diagram and select 'New Class' or 'New Interface'
- Under Package, select an existing package or enter the name of a new one
    - New Packages use dotted notation i.e. 'test.guiTest'
- Enter a suitable class name and click Finish
- Select the desired visibility and modifiers



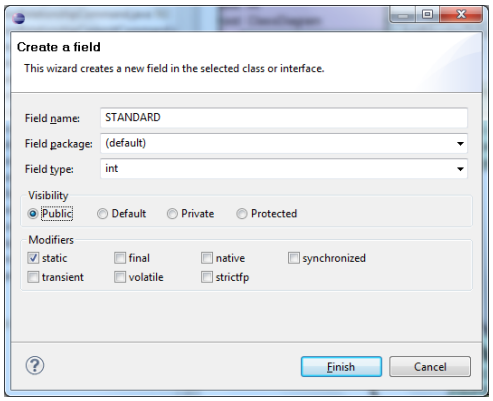
Add a New Method

- Right click on a class and select 'New Method'
- Enter a suitable method name
- Set the return type
    - Either select a type from the list or enter the type
- Choose the visibility and modifiers and if the method is a constructor, check the box
- Add a number of method parameters by:
    - Entering a name
    - Selecting a type (as above)
    - Checking the modifiers
    - Clicking Add
- Click Finish



Add a New Field

- Right click on a class and select 'Create Field'
- Enter a suitable field name
- Select the Field-type's package from the list or enter the name of the package
- Either select a type from the list or enter the type
- Choose the visibility and modifiers
- Click Finish



Modification of Types, Methods and Fields

- All of the settings available in the above wizards can be modified retrospectively
- Select the item by clicking on it and then right click
- To edit the parameters of a Class or Interface select 'Edit Type'
- To edit those of a method, select 'Edit Method'
- To edit those of a field, select 'Edit Field'
- Update the details, including the name if you wish to rename the item
- Click 'Finish' when done

Notice the elements that have been modified have turned blue. This indicates the changes have not yet been written back to the code.

Delete Elements

- Classes and interfaces, methods, fields and relationships can all be marked for deletion:
  - Select the item
  - Right-click on it
  - Click 'Delete Class/Interface' or
  - 'Delete Relationship' or
  - 'Delete Member' (for methods and fields)
- They remain on the diagram, as they are still in the Java code
  - Deleted Classes and Interfaces turn grey
  - Deleted relationships and members turn red
- Once the 'Commit Changes' button is clicked (see below) the element
  - is removed from the diagram and
  - is removed from the code
- Note that you cannot modify a class or member once it has been marked for deletion

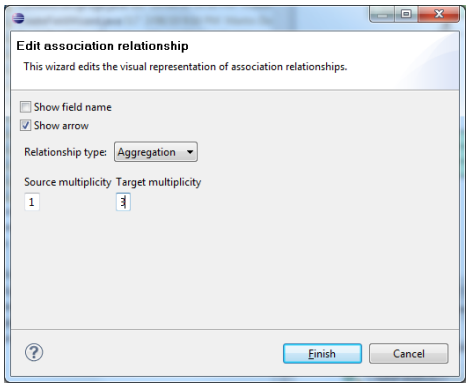
Create a New Relationship

Trip supports creating inheritance, association and implementation relationships. Dependency relationships can also be created but are merely diagram annotations and do not generate any Java code. Association relationships are automatically displayed when a user creates a corresponding field.

- Relationships are added using the palette
- Select the appropriate relationship creation tool
- Click on the child in the new relationship and then click on the parent
- The relationship is shown in UML notation

Association relationships can be annotated with multiplicities and displayed with no arrow, a standard arrow, an aggregation diamond or a composition diamond. These annotations do not generate any Java code.

- Change to the Select tool
- Click on the relationship
- Right click and select 'Edit Association Relationship' and click 'Finish'

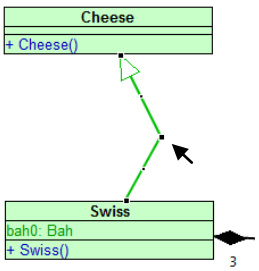


Unfortunately a limitation of the platform prevents self-references from being drawn.

Relationship Routing

While relationships are initially displayed as a straight line, they can be moved around for a more aesthetic presentation.

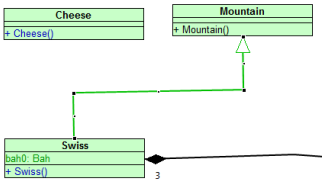
- Select the relationship
- At its centre is a small dot in a white circle
- Click on this bendpoint handle and drag it to position
- This action now creates new bendpoint handles on each side of the initial one
- Remove a bendpoint by straightening the relationship line



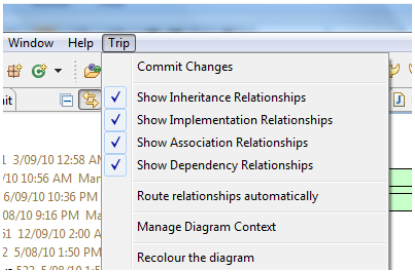
Moving Relationships

Existing relationships can be moved to a different parent or child.

- In Select mode, select the relationship
- Click on the dot at the end of the relationship
- Drag the end to the new target
- The commit process will change the correlating entry in the Java code



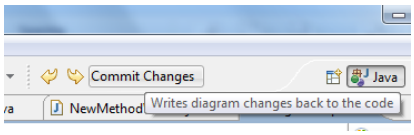
Filter Visible Relationships



Relationships of each type can be hidden from the diagram.

- On the right of the Eclipse menu bar & click the Trip menu
- Use the check boxes to toggle the visibility of all inheritance, association, implementation and dependency relationships

Committing Changes



- Changes on the diagram are committed to code by clicking the 'Commit Changes' button. This can also be accessed in the Trip menu
- A list of changes to be made is presented
- On confirmation, these changes are written to the Java files

Should an error occur during the commit process an error message will appear detailing the problem.

- Click 'Ignore' to disregard the problem and continue processing the transaction
- Click 'Cancel' to reverse the transaction to the state before the 'Commit Changes' button was pressed

Real-Time Mode

As an alternative to the transactional model, Trip offers a real-time mode.

- In the Trip menu, select 'Toggle Real-Time mode'
- Changes made to the diagram are now automatically written to Java code

Saving the Diagram

The diagram can be saved using the normal Eclipse controls and closing the diagram will also prompt for changes to be saved. All uncommitted changes are preserved and when the diagram is loaded, work may resume where it was left off.

Diagram Context

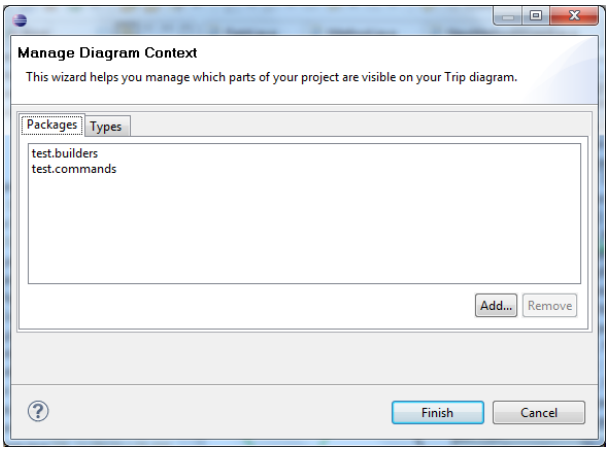
Trip not only generates Java code but also watches for changes in the code and those changes appear on the diagram automatically. Existing classes and packages can be added to the diagram. Their relationships will be discovered and changes made to them in the code will be monitored. In this way the diagram is always up to date with the code.

- To add a class to the diagram, drag it from the Eclipse Project Explorer and onto the diagram.

    - You can select more than one class at a time.
- To add a whole package, drag it from the Project Explorer and onto the diagram.
    - Any new classes created in this package will appear on the diagram.

The content of the diagram can be managed using the Diagram Context feature.

- Select the Trip Menu
- Click 'Manage Diagram Context'



The Packages tab lists all packages that have been explicitly added to the diagram. The Types tab lists all types that have been individually added to the diagram. In this way items on the diagram appear under Packages or Types but not both.

To remove an element or package from the diagram, select the item in the list of Types or Packages and click 'Remove'. This hides the element or package without affecting the Java code.

An alternative to dragging items onto the diagram is to add then using this interface.

- To add a package, select the Packages tab
- Click Add
- Select the package folder from the browser that appears.
- To add a type, select the Types tab
- Click Add
- Select the source file from the file browser that appears.

JUMP TO CODE

To enable quick editing, Trip has the feature whereby the user can jump straight to the place in the code represented by an element on the diagram. There are two ways to do this:

- Double-click on the diagram element or
- Select the element then right-click and choose 'Jump to code'

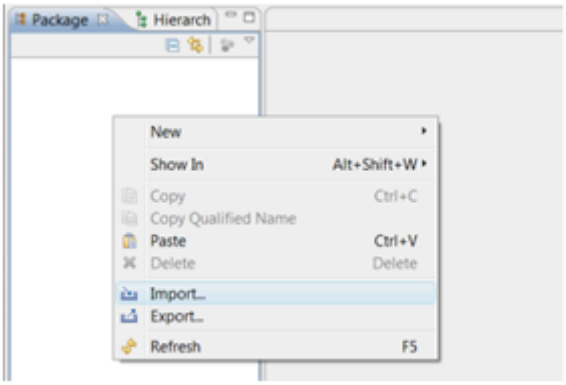
The user can select multiple diagram elements and the jump-to-code feature will display the source files with a warning if more than 10 will be opened.

The Source Code

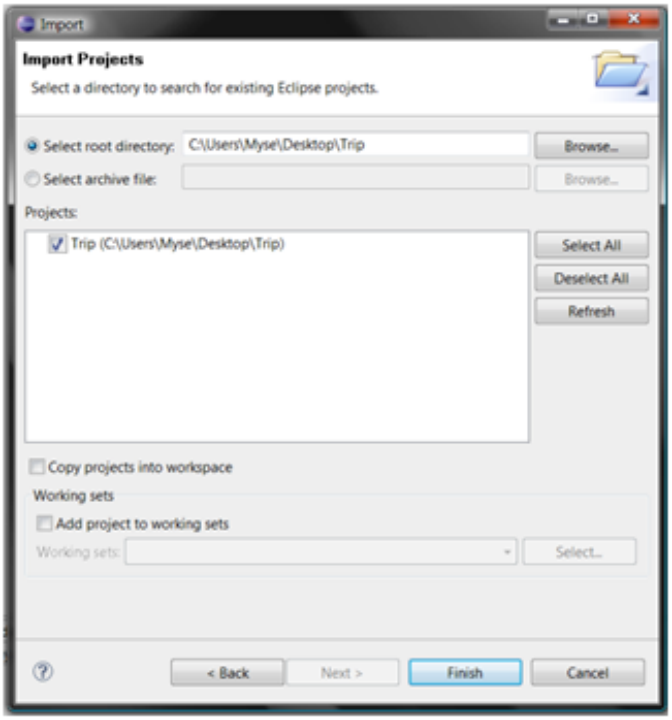
This section is intended to help the reader import the Trip source code and unit tests into Eclipse. It also explains how to build the code. Please note that the Trip Javadoc can be found here.

Importing the Source Code into Eclipse

1. To follow these instructions you will need the Trip.zip file, which can be found on the Final Release page of the wiki (http://redmine.cosc.canterbury.ac.nz:18080/redmine/wiki/2010-c325g3/Final_Release)
2. Unzip Trip.zip, ensuring that the containing folder is called 'Trip'
3. Open Eclipse
4. In the project explorer, right click and select **Import**



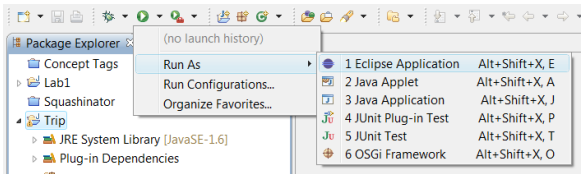
5. Select **Existing Projects into Workspace**
6. Choose the Trip directory and click **Finish**



BUILDING THE SOURCE CODE IN ECLIPSE

To follow these instructions you will need have imported the source code into Eclipse (see Importing the Source Code into Eclipse).

1. Open Eclipse
2. With the Trip project selected, click the arrow next to the Run button and select **Run As>Eclipse Application**

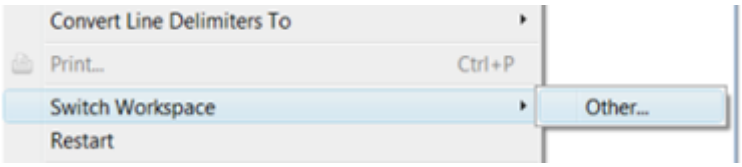


3. Eclipse launches a run-time workspace, which can be used to try out the Trip plug-in

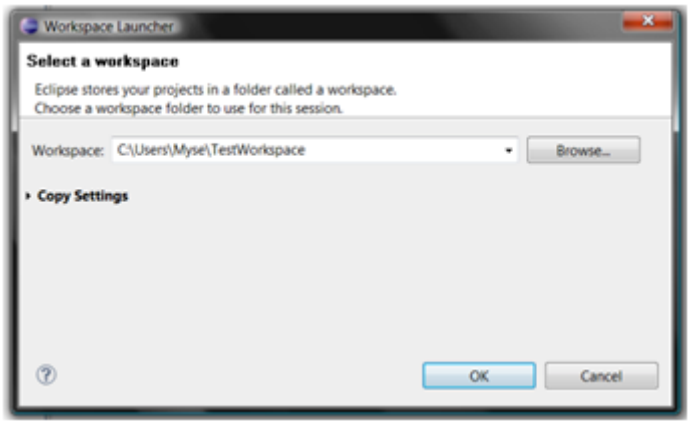
RUNNING TRIP UNIT TESTS

This is a guide to setting up the plug-in test environment. You will need the TestWorkspace.zip file, which can also be found on the Final Release page of the wiki.

1. Unzip TestWorkspace.zip, ensuring that the containing folder is called 'TestWorkspace'
2. Open Eclipse
3. Select **File>Switch Workspace>Other**

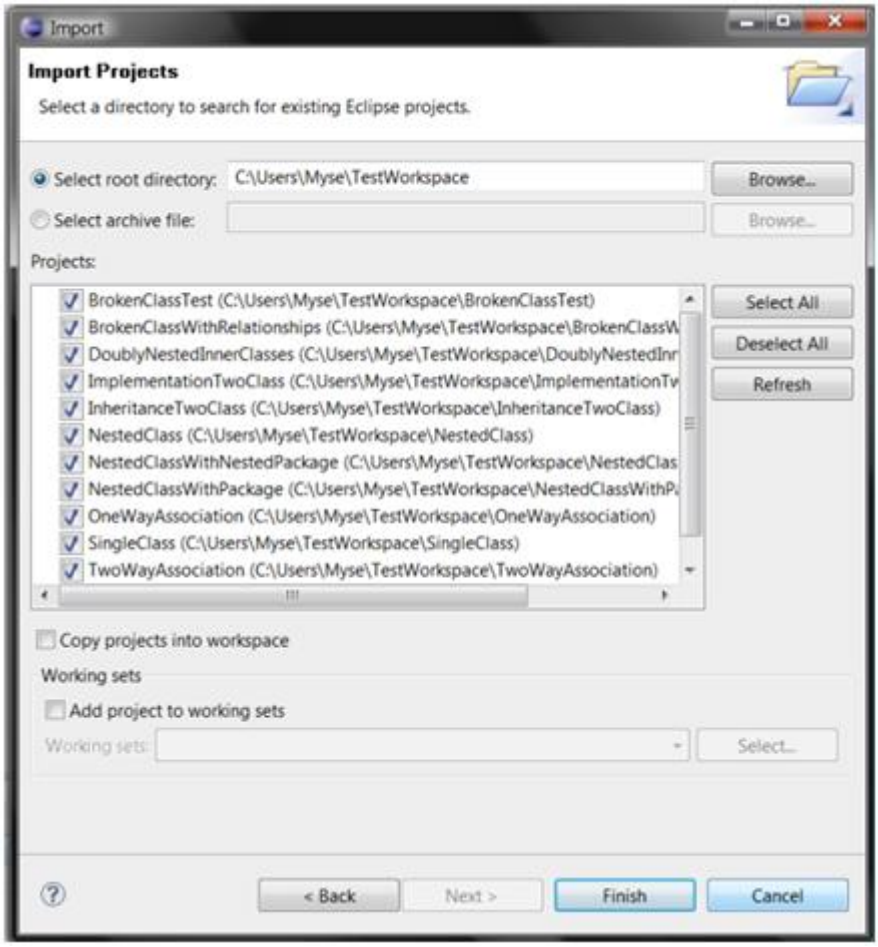


4. In the Workspace Launcher, select your test workspace folder and click **OK**

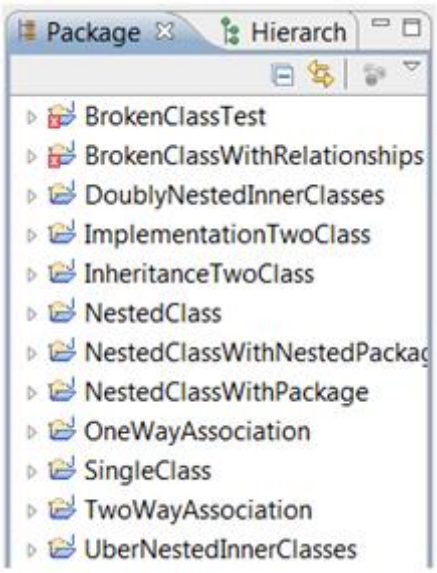


5. In the project explorer, right click and select **Import**
6. Select **Existing Projects into Workspace**

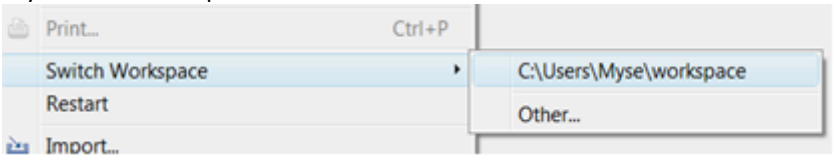
7. Choose the TestWorkspace directory, select all of the projects and click **Finish**



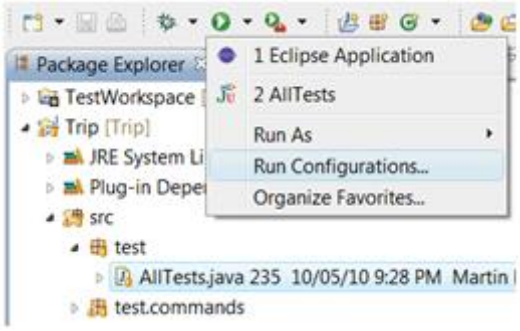
8. Your Project or Package Explorer should now look like this



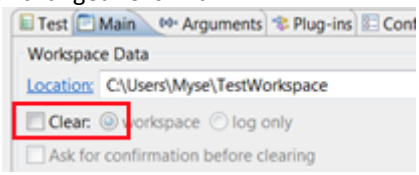
9. Switch back to your main workspace



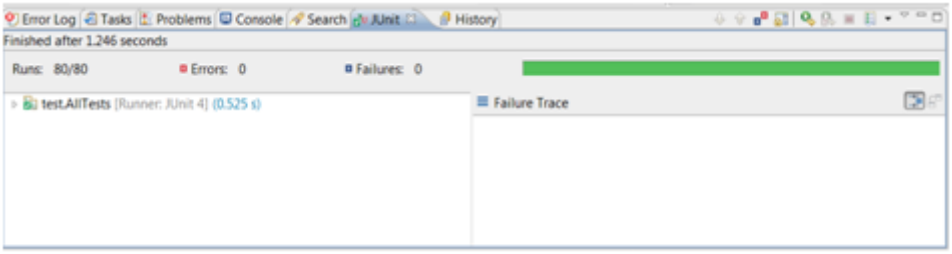
10. While you have the trip.test.AllTests.java file selected, select **Run Configurations** from the **Run** menu



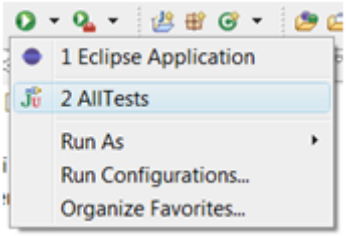
11. Uncheck the **Clear** option and select the root of your test workspace in the **Location** box. The rest of the default options can remain unchanged. Click **Run**



12. If everything is configured correctly, the JUnit view opens in Eclipse



13. To run tests from now on, click the little arrow by the **Run** button and select your run configuration (this was named **AllTests** in this example)



KNOWN ISSUES

- If the user creates a diagram with the same file name and path as an existing diagram, the old diagram is overwritten
- Types on the diagram cannot reference themselves in relationships
- Dependency and association relationships throw an exception on reconnect
- Deleting a method or field and undoing the delete does not restore the method body or field value in code
- There are some synchronization bugs that can occur if the user renames a field or method more than once between commits and then modifies the containing type in the code
- The package list in the new/edit type dialog does not display empty packages
- Generic and array types are not handled properly by the system. The user cannot create parameterized or generic types